AD-A247 571

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

2

2. REPORT DATE

3. REPORT TYPE AND DATES COVERED
FINAL 1 Nov 90 - 30 Apr 91

4. TITLE AND SUBTITLE
"CAIP NEURAL NETWORK WORKSHOP 1990" (U)

5. FUNDING NUMBERS
61102F
2305/B3

6. AUTHOR(S)
R.J. Mammone

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Rutgers-The State University
PO Box 1390
Piscataway, NJ 08855-1089

8. PERFORMING ORGANIZATION REPORT NUMBER
AFOSR-TR- 92 0210

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
Bldg 410
Bolling AFB DC 20332-6448

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFOSR-91-0127

DTIC
ELECTE
MAR 18 1992
S D
D

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for Public Release;
Distribution Unlimited

12b. DISTRIBUTION CODE
UL

13. ABSTRACT *(Maximum 200 words)*

In November 1990, the CAIP Center of Rutgers University organized and hosted a workshop on Neural Networks. The workshop attracted over 120 leaders in the field from the United States and abroad. The goal of the workshop was to assess the current state-of-the-art Neural Network architectectures and algorithms and to consider the most promising directions for further research in this rapidly developing field. A book was printed as an outgrowth of the workshop and constitutes a collection of some of the important papers presented and discussed.

14. SUBJECT TERMS

15. NUMBER OF PAGES
24

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

CAIP NEURAL NETWORK WORKSHOP 1990

AFOSR Final Report

R. J. Mammone
CAIP Center
CoRE Bldg.
Rutgers University
Frelinghuysen Road
Piscataway, NJ 08855-1390

Contents

Weightless Neural Tools: Towards Cognitive Macrostructures
Igor Aleksander

An Estimation Theoretic Basis for The Design of Sorting and Classification Networks
R.W. Brockett

A Self Organizing ARTMAP Neural Architecture for Supervised Learning and Pattern Recognition
Gail A. Carpenter, Stephen Grossberg and John H. Reynolds

Hybrid Neural Network Architectures: Equilibrium Systems That Pay Attention
Leon N. Cooper

Neural Networks for Internal Representation of Movements in Primates and Robots
Rolf Eckmiller, Nils Goerke and Jürgen Hakala

Recognition and Segmentation of Characters in Handwriting with Selective Attention
Kunihiko Fukushima and Taro Imagawa

Adaptive Acquisition of Language
A.L. Gorin, S.E. Levinson A.N. Gertner and E. Goldman

What Connectionist Models Learn: Learning and Representation in Connectionist Networks
Stephen José Hanson and David J. Burr

Early Vision, Focal Attention, and Neural Nets
Bela Julesz

Towards Hierarchical Matched Filtering
Robert Hecht-Nielsen

Some Variations on Training of Recurrent Networks
Gary M. Kuhn and Norman P. Herzberg

Generalized Perceptron Networks with Nonlinear Discriminant Functions

S.Y. Kung, K. Diamantaras, W.D. Mao and J.S. Taur

Neural Tree Networks

Ananth Sankar and Richard Mammone

Capabilities and Training of Feedforward Nets

Eduardo D. Sontag

A Fast Learning Algorithm for Multilayer Neural Network Based on Projection Methods

Shu-jen Yeh and Henry Stark

Weightless Neural Tools: Towards Cognitive Macrostructures

Igor Aleksander
Imperial College of Science, Technology and Medicine
London, UK

1 Introduction

The word "weightless" is used to stress our belief that the future of neural nets lies in understanding the properties of nets with variable function nodes for which the function is loaded by a local algorithm, but not necessarily one constrained by weight variations. The approach dates back to 1965. In 1981, with the advent of inexpensive silicon RAM, it led to the design of an adaptive pattern recognition system called the WISARD (after its designers: Bruce WIlkie, John Stonham, Igor Aleksander, [Recognition Device]). With the current revival of interest in neural computing, it has been possible to show that the RAM approach fully covers the achievements of standard weighted approaches, with the added properties of direct implementability with conventional VLSI techniques and sufficient generality to represent the increasingly complex descriptions of real neurons .

A full overview of weightless neural devices is presented in a companion paper. Here we present a brief description of these systems and concentrate on a fundamental tool we call the General Neural Unit (GNU). This is a flexible associator that can be used as a building brick for advanced neural systems. Specifically, we address interest in neural networks stimulated by Hopfield Aleksander and Hinton et al. which comes from the discovery that a cluster of interconnected neurons has, as an emergent property, the ability to enter stable firing patterns, stimulated by the presentation of parts of these patterns.

An Estimation Theoretic Basis for The Design of Sorting and Classification Networks [1]

R. W. Brockett
Harvard University
Cambridge, Massachussets

1 Introduction

The large and growing literature related to the idea of defining systems of differential equations which, on the basis of incoming stimuli, define categories and assign successive temporal segments to these categories attests to the intrinsic appeal of such an idea. Systems of this type could be used as building blocks in more complex intelligent machines, especially in the lower level "unsupervised" learning portion of such structures. However, because the differential equations which accomplish these tasks are not unique; the choice of a particular system may be difficult to justify. In this paper we show that, in a significant set of cases, algorithms based on the gradient flow equation $\dot{H} = [H,[H,N(u)]]$ can be interpreted as providing a mechanism for computing conditional probabilities. This probabilistic interpretation not only shows that $H(t)$ contains the complete statistical summary of the past stimuli but also provides an interpretation for the undetermined constants which appear.

In [1] we investigated signal processing systems of the form

$$\dot{H} = [H,[H,N_0 + uN_1]]$$

with a view toward using the results previously established for the autonomous system $\dot{H} = [H,[H,N]]$ in this more general setting. Signal processing problems, such as that of generating a quantized version of a signal or that of processing a quantized version of the input with automata-like transformations, were considered explicitly. In order to establish circumstances under which one can interpret equations of this form as being conditional probability propagators, we consider the problem of estimating the state of a finite state stochastic process which is observed with additive white noise. The main new results are to be found in sections [3] and [4] where it is asserted that, under suitable assumptions, this equation does admit such an interpretation and that by suitably choosing the eigenvalues of $H(0)$ we can use this equation as a means for generating a type of associative memory in which the learning and operational aspects are merged.

A Self-Organizing ARTMAP Neural Architecture for Supervised Learning and Pattern Recognition

Gail A. Carpenter†
Stephen Grossberg‡
John H. Reynolds§
Center for Adaptive Systems
and
Graduate Program in Cognitive
and Neural Systems
Boston University

1 Introduction: A Self-Organizing Neural Architecture for Supervised Learning

This chapter describes a new neural network architecture, called ARTMAP, that autonomously learns to classify arbitrarily many, arbitrarily ordered vectors into recognition categories based on predictive success. This supervised learning system is built up from a pair of Adaptive Resonance Theory modules (ART_a and ART_b) that are capable of self-organizing stable recognition categories in response to arbitrary sequences of input patterns. During training trials, the ART_a module receives a stream $\{\mathbf{a}^{(p)}\}$ of input patterns, and ART_b receives a stream $\{\mathbf{b}^{(p)}\}$ of input patterns, where $\mathbf{b}^{(p)}$ is the correct prediction given $\mathbf{a}^{(p)}$. These ART modules are linked by an associative learning network and an internal controller that ensures autonomous system operation in real time. During test trials, the remaining patterns $\mathbf{a}^{(p)}$ are presented without $\mathbf{b}^{(p)}$, and their predictions at ART_b are compared with $\mathbf{b}^{(p)}$.

Tested on a benchmark machine learning database in both on-line and off-line simulations, the ARTMAP system learns orders of magnitude more quickly, efficiently, and accurately than alternative algorithms, and achieves 100% accuracy after training on less than half the input patterns in the database.

ARTMAP achieves these properties by using an internal controller that conjointly maximizes predictive generalization and minimizes predictive error by linking predictive success to category size on a trial-by-trial basis, using only local operations. This computation increases the vigilance parameter ρ_a of ART_a by the minimal amount needed to correct a pre-

dictive error at ART_b. Parameter ρ_a calibrates the minimum confidence that ART_a must have in a category, or hypothesis, activated by an input $\mathbf{a}^{(p)}$ in order for ART_a to accept that category, rather than search for a better one through an automatically controlled process of hypothesis testing. Parameter ρ_a is compared with the degree of match between $\mathbf{a}^{(p)}$ and the top-down learned expectation, or prototype, that is read-out subsequent to activation of an ART_a category. Search occurs if the degree of match is less than ρ_a.

ARTMAP is thus a type of self-organizing expert system that calibrates the selectivity of its hypotheses based upon predictive success. As a result, rare but important events can be quickly and sharply distinguished even if they are similar to frequent events with different consequences.

Between input trials, ρ_a relaxes to a baseline vigilance $\overline{\rho_a}$. When $\overline{\rho_a}$ is large, the system runs in a conservative mode, wherein predictions are made only if the system is confident of the outcome. Very few false-alarm errors then occur at any stage of learning, yet the system reaches asymptote with no loss of speed. Because ARTMAP learning is self-stabilizing, it can continue learning one or more databases, without degrading its corpus of memories, until its full memory capacity is utilized.

Hybrid Neural Network Architectures: Equilibrium Systems That Pay Attention [1]

Leon N Cooper
Brown University

Attitudes toward Neural Networks have, in the short span of my memory, progressed from skepticism through romanticism to what we have at present: general realistic acceptance of neural networks as the preferred - most efficient, most economic - solution to certain classes of problems.

In this brief paper I would like to present an outline of what seem to me to be the major issues and some of the outstanding problems that confront us. In addition, I would like to present a brief account of how our own thinking has progressed

Neural Networks come in several broad categories:

1. Relaxation neural networks that can be regarded as methods of approximating non-linear dynamics.
2. Equilibrium neural networks that classify or assign probabilities
3. Equilibrium hybrid neural networks that via feed-forward and/or feed-back show some properties of relaxation of dynamic networks and display such phenomena as selective attention.

In what follows, we present hybrid equilibrium neural networks that are designed for high efficiency in classification and/or probability ranking and which further have some of the properties of relaxation networks and display selective attention.

Neural networks, in general, do not give optimal solutions. We may regard them in many ways as giving sometimes adequate solutions, sometimes very rapidly. Even training a 3-neuron network as has been shown by Blum and Rivest (Blum 89) is NP-complete. Training a general network is NP-complete, even with only three examples and with two-bit inputs and in some cases they can't even approximate well (Judd 87).

Neural Networks for Internal Representation of Movements in Primates and Robots*

Rolf Eckmiller, Nils Goerke, Jürgen Hakala
Division of Biocybernetics
Heinrich-Heine-Universität Düsseldorf, Germany

1 Introduction

Numerous physiological, behavioral, and theoretical studies in neuroscience suggest topographical arrangements of spatial and temporal information for motor control in various brain regions Especially, the parietal cortex the cerebellum and various parts of the precentral cortex in higher mammals have been implicated in such internal representations of space and spatio-temporal events (trajectories) for motor control.

In contrast to neuroscience, the exploration of neural networks for robot motor control does not require the analysis of existing (biological) systems, but rather the synthesis of technically feasible systems. However, the current knowledge of neuroscience may serve as an important 'concept source'. Afterall, the ability of a fly to generate obstacle-avoiding flight trajectories in a green house in real time or the motor skills of a tennis champion are not based on an algebraic-analytical representation of the various mapping operations as in conventional, software-driven computers for robot control, but on poorly understood geometric-topologically represented functions of dynamic neural networks

This paper compiles recent data on the internal representation of space in the primate oculomotor system and on artificial neural networks for path planning, trajectory storage, and inverse kinematics for the motor control of a planar, redundant robot arm. The contributions exemplify the need to bridge the gap between 'Computational Neuroscience' and 'Neuroinformatics' for the mutual benefit of both neuroscience and computer science.

Recognition and Segmentation of Characters in Handwriting with Selective Attention

Kunihiko Fukushima and Taro Imagawa
Faculty of Engineering Science, Osaka University

Keywords: Neural network, Character recognition, Segmentation, Characters in handwriting, Selective attention.

1 Introduction

Machine recognition of individual characters in handwriting is a difficult problem. It cannot be successfully performed by a simple pattern matching method, because each character changes its shape by the effect of the characters before and behind. In other words, even the same character is scripted differently when it appears in different words, in order to be connected smoothly with the characters in front and in the rear.

One of the authors previously proposed a "selective attention model", which has the function of segmenting patterns, as well as the function of recognizing patterns When a composite stimulus consisting of two patterns or more is presented, the model focuses its attention selectively to one of them, segments it from the rest, and recognizes it. After that, the model switches its attention to recognize another pattern. The model also has the function of associative memory and can restore imperfect patterns. These functions can be successfully performed even for deformed versions of training patterns, which have not been presented during the learning.

We have modified the model and extended its ability to be able to recognize characters in handwriting This paper briefly introduces the new model and offers a preliminary result of computer simulation.

Adaptive Acquisition of Language

A. L. Gorin, S. E. Levinson,
A. N. Gertner and E. Goldman
AT&T Bell Laboratories

1 Introduction

1.1 Problem and Goals

Automated speech recognition (ASR) technology has reached a level of performance such that it is commercially viable for certain carefully chosen applications. However, for even the most elementary of tasks, capabilities fall far short of human performance. We believe that the enormous potential benefit of ASR will not be realized until its performance much more nearly approximates that of humans. Indeed, active research efforts worldwide are aimed squarely at that goal.

Present ASR technology is predicated upon constructing models of the various levels of linguistic structure assumed to compose spoken language. These models are either constructed manually or automatically trained by example. A major impediment is the cost or even the feasibility of producing models of sufficient fidelity to enable the desired level of performance.

The proposed alternative is to build a device capable of acquiring the necessary linguistic skills *in the course of performing its task.* We call this *learning by doing*, and contrast it with *learning by example*. The purpose of this paper is to describe some basic principles and mechanisms upon which such a device might be based, and to recount a rudimentary experiment evaluating their utility for that purpose.

Understanding how to construct such devices would yield valuable technological payoffs. Automated training and on-line adaptation would greatly reduce the human labor required to engineer ASR systems for complex environments. Furthermore, we will see that a system which learns by doing must accept unconstrained input, detect and recover from errors, and then learn from those errors. Such a system must deal with the world as it actually presents itself, rather than how the system designer thought it would be.

What Connectionist Models Learn: Learning and Representation in Connectionist Networks [1]

Stephen José Hanson
Siemens Corporate Research, Inc.
Princeton, NJ

David J. Burr
Bellcore
Morristown, NJ

1 Introduction

There have been historical tensions between the study of learning and the study of representation in psychology and artificial intelligence (AI). For over half a century behavioral psychologists addressed the problem of how knowledge was acquired from experience but ignored the problem of how knowledge and experience were represented internally (Skinner 1950). AI also initially focused on learning (Rosenblatt 1962), but soon turned almost exclusively to the study of representation (Minsky & Papert 1969). With the advent of cognitive psychology (e.g., Miller 1956), internal representation was on psychology's agenda too, but most of the work was still in the style of AI, inspired by the "computer metaphor" (Pylyshyn 1984). Meanwhile, except among behavioral psychologists, the problem of learning was receding into the background. Recently, a new approach, connectionism (Rumelhart & McClelland 1986), has offered not only an alternative "neural network metaphor," but a different style of computation, one that is especially suited to learning and allows the relationship between learning and representation to be studied directly for the first time.

According to popular accounts (Gardner 1985) and their sources (e.g., Miller 1987), the behavioral-to-cognitive shift that began in psychology somewhere around 1956 took place partly because of the gradual pervasion of psychologists' mental and personal lives by computers and partly because none of the available answers to the question "what exactly is learned during learning?" proved to be satisfactory. Psychological phenomena turned out to be too complex to be explained by existing theories. For example, the crucial role of language in human learning far exceeded the explanatory scope of simple learning models, and language learning itself posed uniquely cognitive as opposed to behavioral problems (Skinner 1957, and Chomsky 1959). Yet even the combined resources of behavioral and cognitive psychology have so far proved unable to provide an integrated theory of learning and representation; rather, as in AI, representation is now being studied at the expense of learning.

EARLY VISION, FOCAL ATTENTION, AND NEURAL NETS

Bela Julesz
Laboratory of Vision Research
Psychology Department
Rutgers University
New Brunswick, NJ 08903
and Division of Biology
California Institute of Technology
Pasadena, CA 91125

During the Spring, Summer and Fall of 1990, I wrote five articles on early vision and focal attention for five different disciplines. One manuscript for the ATR Workshop on Modeling Human Visual Perception and Cognition, Kyoto, is intended mainly for engineers in machine vision and AI. The other four manuscripts are as follows: Representations of Vision: Trends and Tacit Assumptions in Vision Research Symposium, (ECVP, Sep. 1990, Paris) entitled Some Strategic Questions in Visual Perception (Julesz, 1991a) is written for my colleagues in psychology. The third is written for physicists, the fourth for neurophysiologists and the fifth for philosophers interested in visual perception. The second article contains about forty strategic questions for vision research in addition to some metascientific ones, and I hope that with these questions I have started a trend and other colleagues will extend my list. The third, and most important one intended for physicists, is an elaborate review entitled Early Vision and Focal Attention, and was written at the request of the Editors of Reviews of Modern Physics (Julesz, 1991b). The fourth article (Julesz, 1991c) of mine is an open peer review in which I answer, among others, the philosopher John Searle on his recent idea that the brain cannot have unconscious processes. This paper will be published sometime next year in the journal Behavioral and Brain Research and I will quote from my answer at a few places here. My fourth article (Julesz, 1990), intended for neurophysiologists, entitled "Early vision is bottom-up except for focal attention," just appeared in Symposium #55: The Brain celebrating the 100th anniversary of the Cold Spring Harbor Laboratory. These five articles span a large audience in different disciplines, and while working on them permitted me to ponder over the state of psychobiology from five different perspectives.

I mention these articles to offer more detailed sources of my views on vision. I will also elaborate here on some of the issues which I discussed at the CAIP Neural Network Workshop on Oct. 16, 1990. I think the interested reader will find in these reviews several topics related to neural networks. Nevertheless, as a token of my appreciation of being invited to this excellent workshop I will make a few brief remarks based on my Kyoto talk, but with some modifications..

During my 32 year career at Bell Laboratories (now at Rutgers University and CALTECH) - doing basic research in visual perception and trying to transfer some of the gained knowledge into technology - I often came across some surprising confusions by leading engineers about the state of brain research. I found it amazing that engineers with excellent brains and great sophistication, usually conservative in assessing technological innovations, would believe in half-baked AI projects. Conversely, there are now interesting technological possibilities to be exploited that for various reasons have not been attempted. A point in case, out of my expertise, is automatic speech recognition, a field whose development I witnessed over a three decades. In spite of great progress, some of the fundamental principles of human speech recognition are still an enigma, and the prodigious feats of a human who can pick out and understand in noisy environment a speaker of his native tongue cannot be mimicked by any machine at present. In visual perception the abilities of some humans are even more enigmatic. Take for instance the art of a cartoonist, who with a few strokes can portray a faithful image of a face that all of us can immediately recognize.

Towards Hierarchical Matched Filtering

Robert Hecht-Nielsen
HNC, Inc.
and
University of California, San Diego

1 Introduction

Traditionally, matched filtering has been used in application areas such as communications, radar, and sonar for detecting a specific waveform in a time series signal. In this paper we examine a new type of matched filter which is optimized for spatiotemporal pattern classification. Banks of these matched filters can be used as high-performance classifiers for spatiotemporal patterns. Unfortunately, the direct implementation of such matched filter banks for large problems (such as large-vocabulary continuous speech recognition), while attractive, is not practical. The focus of this paper is on a method for exploiting the inherent statistical redundancy of typical spatiotemporal pattern sets to allow more efficient implementations of such matched filter banks. In particular, this paper proposes a hierarchical neural network approach to this implementation problem. Before beginning the discussion of generalized matched filtering, some definitions are presented.

For the purposes of this paper a *spatiotemporal pattern* will be taken to be a bounded, continuous function $\mathbf{x} : R \longrightarrow R^n$ of compact interval support from the real numbers (i.e., time) to n-dimensional Euclidean space. Compact interval support implies that the value of the function is the zero vector except in a single closed and bounded interval of time $[a, b]$. Thus, a spatiotemporal pattern is simply a trajectory or path in n-dimensional space, parameterized by time. The set of all such spatiotemporal patterns will be denoted by $P[R, R^n]$.

In spatiotemporal pattern recognition, the typical goal is to provide classifications for relatively brief spatiotemporal patterns (such as words embedded in a continuous speech stream). For simplicity, we will assume that each such brief spatiotemporal pattern belongs to one of M *classes*. For example, in the problem of recognizing words in continuous speech, the input to the system might be a space–time pattern consisting of the time-varying power spectrum of the voltage output of a microphone monitoring the speech of a single speaker. The classes in this instance are the words in the vocabulary. As each word utterance has been completely entered into the speech

classifier system, the system is expected to emit a number between 1 and M, corresponding to the vocabulary number of the word that was spoken. We shall assume that we are dealing with an uncued classification problem without interference and obscuration (i.e., a problem for which there is no significant background noise and only one pattern is present at any time; but for which the patterns appear at unknown times and may abut one another).

Spatiotemporal pattern classification has an issue associated with it that does not pertain to spatial pattern classification — namely, *spatiotemporal warping*. The term spatiotemporal warping refers to the action of a transformation $T : \mathcal{S} \subset P[R, R^n] \longrightarrow P[R, R^n]$ that maps each spatiotemporal pattern in a subset $\mathcal{S}$ of the set of all possible spatiotemporal patterns $P[R, R^n]$ to another spatiotemporal pattern in $P[R, R^n]$. Such spatiotemporal warping transformations can take many forms. One common example is the *time warp*. A time warp takes a pattern $\mathbf{x}(t)$ and transforms it into a pattern $\mathbf{x}(\theta(t))$, where θ is a strictly monotonically increasing smooth scalar function of time. Time warping has the effect of speeding up or slowing down the movement of the pattern $\mathbf{x}$ along its trajectory in R^n and of translating it forward or backward in time.

Another example of a spatiotemporal warp is the change that occurs to the sound power spectrum of a phonograph record when the same record is played at different speeds. In this instance, the spatiotemporal warp transformation is not a simple time warp, because the entire path followed by the spatiotemporal pattern is changed (each sound power feature is changed to a higher or lower frequency channel). Notice that the power spectrum spatiotemporal warping transformation associated with speeding up or slowing down a phonograph record is different from the transformation associated with speaking faster or slower. When we speak faster or slower, our vocal pitch changes very little, and therefore essentially the same sounds are emitted as in normal speech, only in a faster or slower sequence than normal. Thus, the sound power spectrum spatiotemporal warping transformation associated with speaking faster or slower is essentially a simple time warp.[1] Finally, notice that if we were considering the time-series sound signal itself as our spatiotemporal pattern (instead of its power spectrum), these situations would be reversed (speeding or slowing a record would be a time warp, and speaking faster or slower would not be).

In general, spatiotemporal pattern classifiers are required to be insensitive to a general class of spatiotemporal warping transformations. However, in the case where only spatiotemporal warping transformations that are close to the identity transform need to be accommodated, the spatiotemporal pattern can simply be viewed as a fixed

path or curve in n-dimensional space. By dividing the time duration of the pattern into small enough units, we can view the pattern as a sequence of N closely spaced discrete points in n-dimensional space. Alternatively, such a pattern could then be viewed as a single point in an nN-dimensional space. Therefore, at least in principle, a spatiotemporal pattern of finite duration that is not subjected to significant spatiotemporal warping transformations can simply be treated as a spatial pattern. Alternatively, spatiotemporally warped versions of the same pattern can be viewed as *different* spatial patterns (which happen to belong to the same class). If a *time window* of a fixed number N of spatial samples is employed, the total pattern time duration can sometimes be ignored (that is, we can deal exclusively with successive time vignettes — each of which is classified individually as a spatial pattern). This is the approach used by Waibel, et al in their *time-delay neural network*

We now define the generalized matched filter. Following this it is shown how a bank of such matched filters can be used as a classifier for general spatiotemporal patterns that is insensitive to spatiotemporal warping transformations of a given class.

Some Variations on Training of Recurrent Networks

Gary M. Kuhn and Norman P. Herzberg
Center for Communications Research - IDA

1 Introduction

We describe some variations on training of multi-layered recurrent networks which overcome the need for an externally-supplied target function, avoid back-propagation of error derivatives in time, reduce training time, and enhance generalization.

Applied to a speech recognition problem, these variations resulted in as low a number of training iterations and as high a performance, as those reported for cross-entropy trained, hidden Markov models. However, we find that our recurrent networks have *not* provided a large performance improvement over a competing *non*-recurrent network with a similar number of weights.

Generalized Perceptron Networks with Nonlinear Discriminant Functions[1]

S. Y. Kung
K. Diamantaras
W.D. Mao
J.S. Taur

Princeton University
Princeton, NJ 08544

1 Introduction

The objective of this chapter is to provide a systematic exploration of the nonliner perceptron-type networks. For supervised training, the training patterns must be provided in terms of input/output pattern pairs. They are denoted as $[\mathcal{X}, \mathcal{Y}] = \{[\mathbf{x}_1, \mathbf{y}_1], [\mathbf{x}_2, \mathbf{y}_2], \ldots, [\mathbf{x}_M, \mathbf{y}_M]\}$, where M is the number of training pairs.
When applying (both single and multiple layer) supervised neural networks to real applications, two types of basic problem formulations may be adopted:

- *competition-based formulation:*

  Under this formulation, it is not necessary to have the exact values of teachers as direct reference. Instead, the teacher only provide information whether a correct classification is achieved for every training pattern. In other words, the training set $\mathcal{Y}$ will be simply a set of integers labeling the correct class corresponding to each input pattern, i.e. $\mathcal{Y} = \{y_i \in I^1\}$. *The objective of the training is to determine the weights which successfully separate different clusters of patterns and ensure the correct node wins the competition against the rest of the nodes.*

- *approximation-based formulation:*

  Under this formulation, it is assumed that the (exact) values of teachers are available as direct reference. The teacher of the training set $\mathcal{Y}$ will be real-valued N-dimensional vector, i.e. $\mathbf{y}_i = \in R^N$. In other words, corresponding to a specific input pattern each of the N output nodes is assigned a desired value. (This is in a sharp

contrast to the competition-based formulation.) *The objective of the training is to determine the optimal weights which minimize the (least-square) error distance between the teacher's value and the actual response.*

Both formulations lead to very similar mathematical techniques in the actual computations for training. For example, gradients of discriminant functions and back-propagation recursions are useful for both approaches. The main difference is that the approximation formulation can take advantage of having teachers as direct reference while the competition formulation does not need such information.
One can convert a competition formulation into an approximation formulation: Let $y^c = i$ and $\mathbf{y}^a$ denote the teacher values for the competition and approximation formulations respectively. If $y^c = i$, then $\mathbf{y}^a = [-1, -1, \ldots, 1, \ldots, -1]^T$, a vector with all elements being -1 except the ith element which is 1.

Neural Tree Networks

Ananth Sankar and Richard J. Mammone
CAIP Center and Dept. of Electrical Engineering
Rutgers University
Piscataway, NJ 08855-1390

1 Introduction

Pattern classification is a fairly mature subject and there have been many books written on this field The supervised pattern classification problem can be stated as: Given a set of training feature vectors, $\mathbf{X}_i$, each with an associated class label $\mathbf{Y}_i$, find a retrieval system that will produce the correct label , $\mathbf{Y}_i$ for any feature vector, $\mathbf{X}_i$ The retrieval system is determined by using a training algorithm. In recent years there has been increased interest in the use of neural networks as pattern classifiers and associative memories Feedforward neural networks, in particular, have emerged as one of the most successful neural network architectures. The basic building block for feedforward neural networks is a neuron which calculates a non-linear function of the weighted sum of its inputs. Thus the neuron can be specified by its weights and the non-linear activation function. In a feedforward neural network these neurons are arranged in layers such that the outputs of one layer are connected to the inputs of the next layer. The input feature vector is fed to the first layer of neurons whose outputs become the inputs for the second layer of neurons and so on and finally the output of the last layer is the output of the network. The last layer is called the output layer and all other layers are called hidden layers. An n-layer network is one that has $n-1$ hidden layers and one output layer. The weights of such systems are typically found by supervised training algorithms that use a training set of labeled feature vectors. The problem of training a feedforward network is NP-complete and therefore one must look for good heuristic solutions. Currently the most popular heuristic is backpropagation which is essentially a gradient descent method over an error surface. For networks with hidden layers, this error surface is non-quadratic, and creates problems with local minima. In addition, the exact number of hidden neurons and the connectivity

between layers must be specified before learning can begin. In practice, however, one cannot guarantee that backpropagation will find the correct weights for a given number of neurons and a particular training set. The most common solution to this problem is to choose the number of hidden neurons by trial and error.

Decision trees provide another popular approach to pattern classification The structure of a decision tree is recursive in that each node of a decision tree has a set of child nodes, each of which is also a decision tree. The terminal nodes have no child nodes and are called leaf nodes. It has been shown that constructing a decision tree with the shortest length to solve a given classification task is NP-complete Thus all existing algorithms to grow decision trees are heuristic methods. The essential idea in these algorithms is to solve the problem by using a divide and conquer approach. The root of the tree partitions the feature space into subsets, assigning each subset to a child node. This partitioning is also called splitting. The usual way to split the feature space is to generate a list of possible splits and then search through this list to find the best split This splitting process is continued until each terminal or leaf node corresponds to one class. Different ways of generating the search space of partitions and evaluating the "goodness" of a split lead to different decision tree algorithms like CART [4] and ID3 These algorithms require an exhaustive search through a list of arbitrarily generated splits. This search process is computationally inefficient. In addition, since the list of possible splits is generated in an adhoc manner, the solution may not be close to the optimal solution. Most decision tree algorithms use splits which result in regions whose boundaries are perpendicular to the feature space axes. This is a severe limitation in cases where the problem is linearly separable but the decision hyperplane is not perpendicular to a feature axis since many perpendicular splits would be needed to approximate the hyperplane. CART allows for non-perpendicular splits but this, too, involves an exhaustive search technique.

In this paper we introduce a new neural network architecture which is a combination of feedforward neural networks and decision trees. The architecture is called a Neural Tree Network (NTN). The NTN architecture is a tree with a single layer neural net at each of its nodes. The new architecture is grown during the learning process rather than specified a priori as in feedforward neural networks. We show that the NTN architecture offers a substantial implementation advantage over feedforward neural networks. A new learning

algorithm which grows the NTN is also described. The new learning algorithm is based on the method of gradient descent and does not require an exhaustive search as previously used in training algorithms for decision trees. The splits are not restricted to be perpendicular to the feature space axes and the new algorithm is guaranteed to converge to a solution. Simulation results show that the new algorithm is faster than backpropagation and produces smaller trees than conventional decision trees.

This paper extends the results of our earlier work on two class problems to the multi-class case. The paper is organized as follows. Section 2 discusses the new neural net architecture. In section 3, we describe the new algorithm that grows the NTN. In the discussion of the new architecture and algorithm, we concentrate on the two class problem for the sake of simplicity. In section 4, we extend the new method to multi-class pattern recognition problems. Section 5 shows simulation results and compares the new algorithm to backpropagation and decision trees and section 6 gives the summary and conclusions.

Capabilities and Training of Feedforward Nets

Eduardo D. Sontag
Rutgers University

1 Introduction

This paper surveys recent work by the author on learning and representational capabilities of feedforward nets. The learning results show that, among two possible variants of the so-called backpropagation training method for sigmoidal nets, both of which variants are used in practice, one is a better generalization of the older perceptron training algorithm than the other. The representation results show that nets consisting of sigmoidal neurons have at least twice the representational capabilities of nets that use classical threshold neurons, at least when this increase is quantified in terms of classification power. On the other hand, threshold nets are shown to be more useful when approximating implicit functions, as illustrated with an application to a typical control problem.

A Fast Learning Algorithm for Multilayer Neural Network Based on Projection Methods

Shu-jen Yeh
Rensselaer Polytechnic Inst.

Henry Stark
Illinois Inst. of Technology

1 Introduction

Artificial neural nets shows great promise as associative memories and pattern classifiers As is well known, the single layer perceptron can yield good results as pattern classifier when the classes are separable by hyperplanes but fails when this condition is not met. Thus the exclusive-or problem cann be solved by a single layer neural net nor, for example, can the classification of classes distinguished by distance-from-the-origin On the other hand, many problems of practical interest can be solved by two-layer neural nets, including the ones cited above. To train a multi-layer neural net there exist a number of learning algorithms of which the back-propagation learning rule (BPLR) is the most popular. The BPLR is an iterative gradient search algorithm designed to minimize the mean-square error between the actual output of a multi-layer feed-forward net and the desired output. It is also well known that the BPLR shows good performance in learning but exhibits slow learning speed in many cases.